BrainGoated

CMPT 276 - D200: Introduction to Software Engineering

Manjari Prasad

Alisha Mahebub Jesani

Gurpreet Kaur

Beyzanur Kuyuk

GitHub Repository : final-project-11-stars

https://github.com/CMPT-276-SPRING-2025/final-project-11-stars.git

Overview

Despite the massive amount of educational content available online, most of them rely on traditional methods which lack in offering engaging material more interactively. Most of the engaging and interactive content either lacks customisation options or is not specifically designed with children and teenagers in mind. Our website aims to resolve that by creating quizzes specifically catered towards ages 8-16 while allowing them to choose a category of their interest. In addition to that, parents and teachers also have an avenue to offer their children an interactive learning tool to teach key concepts enjoyably.

We brainstormed multiple ideas for this project and after a lot of consideration, we ultimately decided on a quiz website- BrainGoated- specifically tailored towards kids aged 8-16. We reflected on existing learning platforms to identify their shortcomings. They are either too generic or not interactive enough for children to be interested in them. Our website will provide a fun and interactive way for children to put their knowledge to the test across various categories while maintaining a kid-friendly design.

Primary Users

Children (aged 8-16):
Need: Engaging quizzes that align with their interests, along with an opportunity to ask follow-up questions on topics regarding the quiz.

Solution: Our website will offer subject-based quizzes with interactive features, a chatbot to assist them with any further questions they might have, rewards, and adaptive difficulty levels to maintain engagement.

Parents:
Need: A simple, safe and reliable learning tool for kids which keeps them engaged and knowledgeable.

Solution: Our website will offer an easy-to-use method to ensure parents are concern-free over time-consuming aspects like account creation or privacy while providing an interactive platform.

Teachers/Educators:
Need: A tool for them to verify their students' understanding of concepts taught in class.

Solution: Our website will offer an avenue for teachers to create and direct students to quizzes which align with what is being taught in class.

User Personas

1. Student Persona:

Name: Neeraj
Age: 9
Background: Neeraj is a curious 4th grader who loves to explore new topics such as history, space, math, nature and coding and test his knowledge. He loves to ask questions and has a never-ending thirst for knowledge. Neeraj does not like learning through standard methods such as textbooks and prefers interactive tools and quizzes to test his knowledge. He likes to see how well he did as he is competitive and pushes himself to unlock his potential.

Goals:
- Have fun while learning through engaging activities.
- Be able to ask questions while learning the material and receive immediate responses.

Frustrations:
- Does not like the traditional way of learning from textbooks
- Finds it difficult to get a one-on-one learning experience on specific topics he is interested in.

Needs:
- Fun, game-like trivia quizzes about various categories and topics that he is passionate or curious about.
- Being able to ask any questions he has regarding his interests and asking follow-up questions about what he has learned.

2. Parent Persona:

Name: Seeja
Age: 41
Background: Seeja is a working mom of 2 kids, one of which is Neeraj. She wants her kids to pursue their interest in various topics and stay engaged while learning. Her kids are very curious and love to ask questions about different topics they are interested

in. She has difficulty answering all their questions and would love to provide the knowledge to quench their curiosity.

Goals:

- Find a reliable, age-appropriate platform where her kids can have engaging learning experiences.
- Ensure her children have access to accurate and educational answers to their questions.
- Keep her kids engaged in productive, curiosity-driven activities rather than passive screen time.

Frustrations:

- She wants her kids to spend time learning and discovering their interests instead of spending time on social media platforms.
- Hard to find platforms that would be interesting or catered to children's learning.

Needs:

- A platform catered toward kids which offers a fun and interactive learning experience.
- Easy access to various topics kids can explore and test their knowledge on.

3. Teacher Persona:

Name: Mr. Bob

Age: 45

Background: Mr. Bob is a grade 8 Science teacher who likes to provide his students with tools that would get them to interact with the material being taught. He wants his students to be proactive in their learning and discover new knowledge related to the material. He believes that students learn best when they actively participate, explore new concepts, and receive immediate feedback on their understanding.

Goals:

- Find a platform for kids to explore new topics and discover interests related to class material.
- Provide an alternative way of learning to the traditional textbook method of education.
- Offer students a fun and competitive way to test their knowledge.

Frustrations:

- He finds it difficult to get his students to engage with the material and provide a more fun learning experience.

- Limited time to answer every student's personal questions about the material while managing the entire class.

Needs:

- A safe platform for students to learn and further explore their interests aside from what is being taught during class.
- A chatbot assistant that can help answer student questions, freeing up his time for deeper discussions.
- A platform that allows teachers to create custom quizzes for their students.

Chosen API's

1. The Trivia API – This API provides a wide range of quiz questions across different topics and difficulty levels. It assists in creating a dynamic and engaging trivia experience by offering diverse question formats.

2. OpenAI API – It enables AI-powered features such as generating personalised learning materials and explanations. Additionally, it enhances the quiz experience by providing intelligent insights and interactive content.

   (Backup APIs)

3. Google Text-to-Speech API –This API converts text into life-like speech, making the quiz more interactive and accessible. It enhances engagement by reading quiz questions aloud and providing fun voice effects for customization.

4. Open Trivia API – This API provides access to a vast collection of trivia questions categorized by topic, difficulty level, and question type. It allows developers to integrate engaging and customizable quiz content into their applications.

API Features

1. **The Trivia API**

   Features:

- Category-Based Questions – Users can choose trivia questions from various predefined categories.

- Quiz Question Generation – Creates a set of quiz questions dynamically based on user preferences.

- Join quiz with ID – Gives the creator of the quiz a unique ID. Inputting it gives access to their quiz.

2. **OpenAI API**

   Features:

- Enhanced Learning Data – Provides additional learning resources and explanations related to the quiz topic.

- Interactive Chatbot – Enables users to ask an AI chatbot for clarifications, hints, or topic-related discussions.

- AI-Generated Quiz – Dynamically creates quiz questions using AI, offering a more personalized quiz experience.

3. **Open Trivia API**

   Features:

- Extensive Question Bank – Provides a large collection of trivia questions categorized by topic.
- Difficulty Level Selection – Users can choose between easy, medium, and hard questions.
- Randomized Quiz Questions – Ensures variety by fetching different questions each time a quiz is started.

4. **Google Text-to-Speech API (For Interactive Audio)**

   Features:

- Read-Aloud Mode – Converts quiz questions and answer choices into speech so kids can listen instead of just reading.

- Supports – Adds light background music and fun sound effects (e.g., a chime for correct answers, a buzzer for wrong answers) to make the quiz more immersive and engaging for children.

- Voice Effects – Allows kids to choose different voices (e.g., robot, storyteller, or cartoon character) for an immersive experience.

User Stories:

1) **Trivia API:**

- Category-Based Questions

  As a young learner, I want to choose a quiz category, so that I can answer questions based on my interests.

- Quiz Generation

  As a teacher, I want to create a quiz for my students, so that they can interactively engage with the lecture material.

- Join the quiz with ID

  As a teacher, I want to generate a shareable quiz ID, so that my students can join the same quiz and compete.

2) **Open AI API:**

- Enhanced Learning Data

As a student, I want to receive additional learning material before a quiz, so that I can better understand the topic and improve my performance.

- Interactive Chatbot

  As a quiz participant, I want to ask an AI chatbot for help when I get a question wrong so that I can learn from my mistakes and improve my knowledge.

- AI-Generated Quiz

  As a quiz creator, I want to generate a quiz based on a topic of my choice, so that I can play with my friends or test my knowledge.

3) **Google Text-to-Speech API (For Interactive Audio)**

- Read-Aloud Mode

  As a user with reading difficulties, I want to enable a read-aloud feature, so that I can listen to questions and answers instead of reading them.

- Voice Effects

  As a quiz participant, I want to change the voice reading the quiz questions, so that I can make the experience more fun and engaging.

- Language Translator
  As a language learner, I want to hear quiz questions in my target language, so that I can improve my comprehension skills while playing.

4) Open Trivia API

- Extensive Question Bank

As a student, I want access to a wide variety of quiz questions across multiple topics so that I can explore my diverse interests and test my knowledge. However, it is difficult to find a single platform that gathers all my interests in one place, allowing me to engage with quizzes on everything I enjoy.

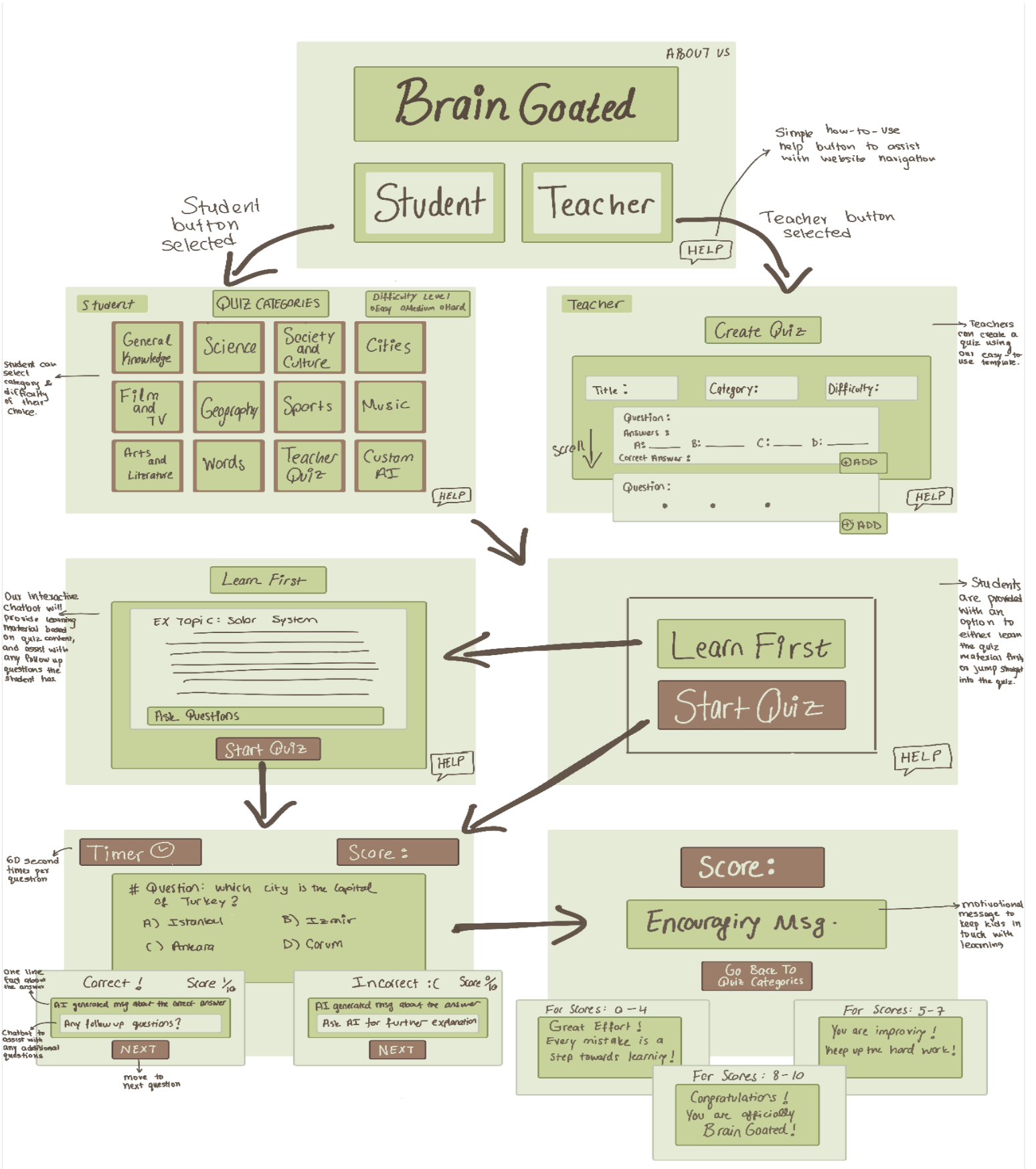
- Difficulty Level Selection – Users can choose between easy, medium, and hard questions.

  As a student, I want to select the difficulty level for quiz questions (easy, medium, or hard) so that I can start with simpler questions to build confidence and gradually challenge myself as I learn more.

- Randomized Quiz Questions – Ensures variety by fetching different questions each time a quiz is started.

  As a quiz enthusiast, I want to explore a variety of new questions based on my subjects, so that I don’t have to repeatedly use the same quiz questions from my textbook.

Low-fidelity Storyboard



Chosen Front-End Technology Stack

1. React.js (Framework) – For Structure & Logic

- React will contribute to the structure and logic of the website, as it is highly recommended for handling dynamic content like trivia questions, feedback, and interactive elements.
- It is well-suited for building single-page applications (SPAs), providing a smooth user experience, which is great for maintaining real-time quiz interactions and leaderboards.
- React has a built-in navigation tool called React Router, eliminating the need for a backend to manage page transitions.
    - React Context API (State Management)
        - The React Context API will be used to manage the global state without needing a backend or external libraries like Redux.
        - It allows storing and accessing key data (e.g., quiz progress, scores, difficulty settings) across components without prop drilling.
        - This ensures seamless updates to the UI when users answer questions, complete quizzes, or interact with leaderboards.

2. Tailwind CSS & CSS Modules (Styling & Animations)

- We will use Tailwind CSS for rapid, responsive styling and a clean design, ensuring a modern look.
- To make the app kid-friendly, we will implement CSS Modules for component-level styling, allowing for fun animations, colourful UI, and engaging transitions.
- Tailwind's utility-first approach will keep the styling minimal, reducing complexity while maintaining customization options.

3. Axios (API Handling)

- We will use Axios to efficiently fetch trivia questions, AI-generated responses, and fun translations from external APIs.
- It simplifies error handling and response transformation, making API interactions seamless.